

## Remote booting of an operating system by a network

Patent Number:  US5577210

Publication date: 1996-11-19

Inventor(s): ABDOUS ARAVE (FR); DALONGVILLE DIDIER (FR); DEMORTAIN STEPHANE (FR)

Applicant(s): BULL SA (FR)

Requested Patent:  EP0538133, B1

Application Number: US19930075578 19930616

Priority Number (s): FR19910012837 19911017; WO1992FR00977 19921015

IPC Classification: G06F9/445

EC Classification: G06F9/445B8

Equivalents: DE69228413D, DE69228413T, ES2129439T,  FR2682786, JP2590401B2, JP5509430T, SG54242,  WO9308527

---

### Abstract

---

PCT No. PCT/FR92/00977 Sec. 371 Date Jun. 16, 1993 Sec. 102(e) Date Jun. 16, 1993 PCT Filed Oct. 15, 1992 PCT Pub. No. WO93/08527 PCT Pub. Date Apr. 29, 1993A method for remote booting by a server of at least one terminal, including a volatile memory (RAM), at least one processor, a telecommunications card enabling connection to the server by a network, said server being provided with telecommunication device and memory of sufficient capacity to store the communications protocol, the operating system of the terminals recopied into an image file, its own operating system, and the applications programs, wherein the method includes the remote loading of a startup program by transforming the first interruption produced by a terminal following its being powered up into a request for reading an image file memorized in the server.

---

Data supplied from the esp@cenet database - I2



⑯ BUNDESREPUBLIK  
DEUTSCHLAND



DEUTSCHES  
PATENT- UND  
MARKENAMT

⑯ Übersetzung der  
europäischen Patentschrift

⑯ Int. Cl.<sup>6</sup>:  
G 06 F 9/445

⑯ EP 0 538 133 B 1

⑯ DE 692 28 413 T 2

- ⑯ Deutsches Aktenzeichen: 692 28 413.3  
⑯ Europäisches Aktenzeichen: 92 402 820.2  
⑯ Europäischer Anmeldetag: 15. 10. 92  
⑯ Erstveröffentlichung durch das EPA: 21. 4. 93  
⑯ Veröffentlichungstag der Patenterteilung beim EPA: 10. 2. 99  
⑯ Veröffentlichungstag im Patentblatt: 24. 6. 99

⑯ Unionspriorität:  
9112837 17. 10. 91 FR

⑯ Patentinhaber:  
Bull S.A., Puteaux, FR

⑯ Vertreter:  
Prinz und Kollegen, 81241 München

⑯ Benannte Vertragstaaten:  
DE, ES, FR, GB, IT

⑯ Erfinder:

Abdous, Arave, F-91300 Massy, FR; Dalongville,  
Didier, F-91940 Gometz le Chatel, FR; Demortain,  
Stephane, F-91120 Palaiseau, FR

⑯ Fernladung eines Betriebssystems über ein Netzwerk

Anmerkung: Innerhalb von neun Monaten nach der Bekanntmachung des Hinweises auf die Erteilung des europäischen Patents kann jedermann beim Europäischen Patentamt gegen das erteilte europäische Patent Einspruch einlegen. Der Einspruch ist schriftlich einzureichen und zu begründen. Er gilt erst als eingelebt, wenn die Einspruchsgebühr entrichtet worden ist (Art. 99 (1) Europäisches Patentübereinkommen).

Die Übersetzung ist gemäß Artikel II § 3 Abs. 1 IntPatÜG 1991 vom Patentinhaber eingereicht worden. Sie wurde vom Deutschen Patent- und Markenamt inhaltlich nicht geprüft.

DE 692 28 413 T 2

DE 692 28 413 T 2

Die vorliegende Erfindung betrifft ein Verfahren zum Fernladen eines Betriebssystems über ein Netz. Aus der britischen Patentanmeldung 2 231 180 ist bekannt, daß Laden eines ersten Rechners, der über ein Netz mit einem zweiten Rechner verbunden ist, auszuführen, indem eine erste Untereinheit des Betriebssystems in den Speicher des ersten Rechners geladen wird, wobei die Untereinheit die Befehle enthält, die das Kopieren der Dateien, das Anlegen eines Verzeichnisses, das Formatieren der Platte sowie den Betrieb einer Verbindung über das Netz, um die Gesamtheit der Dateien des Betriebssystems auf die Platte des ersten Rechners zu übertragen, ermöglichen. Das Verfahren dieser Patentanmeldung weist den Nachteil auf, daß es ein erstes Laden einer Untereinheit des Betriebssystems in den Speicher und anschließend das Vorhandensein eines Magnet-Peripheriegeräts wie etwa einer Festplatte oder eines Diskettenlesegeräts im ersten Rechner, auf die das Betriebssystem geladen wird, erfordert. Dieser Prozedurtyp ist vor allem mit dem Fernladen von Terminals unvereinbar, das in einer sehr kurzen Zeit ausgeführt werden muß.

Es ist außerdem aus dem amerikanischen Patent 4 958 278 ein Verfahren zum Laden von Daten oder Programmen auf mehrere Terminals bekannt. Diese Terminals sind nicht mit Diskettenlesegeräten versehen. Sobald eine Anforderung zum Laden durch eine der Stationen nach ihrem Einschalten gesendet wird, schickt der Server den ersten Rahmen, wartet auf die Quittierung und schickt bei Empfang der Quittierung die nachfolgenden Rahmen. Wenn im Verlauf des Sendens der Rahmen, das dem Laden einer ersten Station entspricht, ein weiteres Terminal eine Anforderung zum Laden sendet, übernimmt die zweite Station die für die erste Station bestimmtem gesendeten Rahmen, wobei sie die Tatsache, daß die durch die erste Station empfangenen ersten Rahmen fehlen, in einer Tabelle notiert. Jede Station muß nach dem Senden eines Rahmens ihre Quittierung schicken. Wenn alle für die erste Station bestimmten Rahmen gesendet

worden sind, empfangen die Stationen, die sich später an den Speicher geschaltet haben, die ersten Rahmen, die sie aufgrund ihres späteren Dazuschaltens nicht übernehmen konnten. Bei diesem Verfahren speichert der Server mit dem Auftreten einer zweiten Station im Speicher die Nummer des ersten Rahmens, den diese Station empfängt, im Speicher und schickt am Ende des Ladens der ersten angeschlossenen Station an die andere Station die Rahmen, die diese nicht empfangen konnte. Dieses Verfahren verpflichtet jede Station zum Senden ihrer Quittierung, was unterschiedliche Quittierungssendezeiten für jede Station notwendig macht, so daß Kollisionen vermieden werden. Dies bringt eine Verzögerung zwischen dem Ende des Sendens eines Rahmens und dem Senden des nächsten Rahmens mit sich. Diese Verzögerung muß entweder abhängig von der Anzahl der angeschlossenen Stationen variabel sein oder einer Maximalkonfiguration entsprechend berechnet sein.

Jedenfalls ermöglichen diese Vorrichtung und dieses Verfahren keinen starren Betrieb gemäß einem bestimmten Programm zum Fernladen, das das Laden einer festgelegten Anwendung ein für allemal ausführt.

Aus dem europäischen Patent 0 358 282 ist ein Verfahren zum Fernladen eines Betriebssystems in Terminals bekannt. Die Platten der Terminals sind für dieses Verfahren nicht notwendig. Dieses Verfahren besteht darin, in die Initialisierungsroutine der Speicherkommunikationskarte eine Anforderung an die Speichereinheit einzuführen, die eine Ureingabe anfordert, die geschickt wird, sobald der Selbsttest der Busadressen des Terminals das Vorhandensein der Karte erfaßt. Der Server antwortet dann auf diese Anforderung, indem er ein minimales Ureingabeprogramm zum Terminal überträgt, das dieses Programm in seinen Speicher lädt. Dieses Programm ermöglicht anschließend das Management der durch die Fortsetzung der Ureingabe der Station bedingten Unterbrechungen, wobei eine Verknüpfung der Unterbrechungen mit einer Zugriffsanforderung auf eine Serverdatei erzeugt wird. Dieses Verfahren weist den Nachteil auf, daß es zuerst ein Programm in den Terminalspeicher fern-

lädt, was einen Zeitverlust hervorruft und im Speicher der Station den verfügbaren Platz verringert.

Ein Ziel der Erfindung ist, ein Verfahren für das Laden von Terminals, die keine Platten- oder Disketteneinheit enthalten, vorzuschlagen, das für eine Architektur aus Standardterminals, die mit dem PC-Standardbetriebssystem (MSDOS) betrieben werden, das spätere Laden aller Anwendungen oder benötigten Dateien ermöglicht.

Dieses Ziel wird erreicht durch die Tatsache, daß das Verfahren zum Fernladen wenigstens eines Terminals durch einen Server, wobei das Terminal einen Schreib-Lese-Speicher (RAM), wenigstens einen Prozessor und eine Telekommunikationskarte enthält, die den Anschluß an den Server des Netzes ermöglicht, wobei der Server mit Telekommunikationsmitteln und mit Speichermitteln mit einer Kapazität versehen ist, die ausreicht, um das Kommunikationsprotokoll, das Betriebssystem der Terminals, das in eine Bilddatei kopiert ist, sein eigenes Betriebssystem und die Anwendungsprogramme zu speichern, dadurch gekennzeichnet ist, daß es eine Prozedur zum Fernladen eines Ureingabeprogramms enthält, die darin besteht, die erste Unterbrechung zum Lesen des Ureingabeblocks auf der lokalen Platte, die von einem Terminal erzeugt wird, wenn es unter Spannung gesetzt wird, in eine Anforderung zum Lesen einer im Server gespeicherten Bilddatei umzusetzen, wobei das Programm zum Umsetzen von Unterbrechungen (EMUL INT) in einem Festwertspeicher (PROM) der Kommunikationskarte gespeichert ist.

Gemäß einem weiteren Merkmal besteht der Schritt des Fernladens des Ureingabeprogramms darin, in den Schreib-Lese-Speicher RAM den Ureingabesektor (BOOT) zu laden, die Ausführung dieser Ureingabedatei zu starten und die Unterbrechungen der Zugriffsanforderungen auf Dateien von Magnet-Peripheriegeräten, die nicht im Terminal vorhanden sind, in Anforderungen zum Lesen der entsprechenden Bilddatei im Server durch eine Koordinatentransformationsprozedur umzusetzen.

Gemäß einem weiteren Merkmal besteht die Koordinatentransformationsprozedur darin, die Koordinaten physischer Sektoren von Dateien von Magnet-Peripheriegeräten in eine logische Kapitelnumerierung der Bilddatei, die dem Sektor der aufgerufenen Datei entspricht, durch eine Sektorentsprechungstabelle umzusetzen, die Anforderung zum Lesen des Kapitels an den Server zu übertragen und die Kapitel der Bilddateien an sämtliche Terminals zu senden.

Gemäß einem weiteren Merkmal enthält der Schreib-Lese-Speicher RAM des Terminals nach dem ersten Fernladen des Ureingabeprogramms eine Dateizuordnungstabelle (FAT), die eine Korrespondenz zwischen der Diskettennummer, der Kopfnummer, der Spurnummer, der Spursektornummer und dem Dateinamen herstellt, sowie eine Sektorbeschreibungstabelle (SDT), die eine Entsprechung zwischen der Diskettennummer, der Spurnummer, der Spursektornummer und der Kapitelnummer der Bilddatei des Servers herstellt.

Gemäß einem weiteren Merkmal ermöglicht das Verfahren das Fernladen von Dateien des Betriebssystems, das durch die Dateien IO.SYS, MSDOS.SYS, CONFIG.SYS und COMMAND.COM gebildet ist.

Gemäß einem weiteren Merkmal ist die Dateizuordnungstabelle (FAT) mit dem Dauerstatus (KEEP FOR EVER) im Schreib-Lese-Speicher RAM gespeichert, wobei dieser Status durch ein Bit des Zustandsworts der Sektorbeschreibungskarte angegeben wird.

Ein weiters Ziel ist, ein Verfahren zum Fernladen mehrerer Terminals vorzuschlagen, das die Wartezeit der Terminals begrenzt.

Dieses Ziel wird dadurch erreicht, daß der Server dann, wenn ein Terminal eine Anforderung eines ersten Kapitels zu diesem Server schickt, eine Zeit T1 vor seiner Antwort abwartet, um sicherzustellen, daß die anderen Terminals nicht die gleiche Anforderung haben.

Gemäß einem weiteren Merkmal wird eine Anforderung des ersten Kapitels durch eine andere Station, die vor dem Ablauf einer Zeit T2 nach der Anforderung desselben ersten Kapitels durch eine erste Station ankommt, sofort berücksichtigt, wobei die Fernladeprozedur für die Gesamtheit der Stationen, die Anforderungen gesendet haben, für diese Stationen auf null zurückgesetzt wird.

Gemäß einem weiteren Merkmal schließt der Server die Sitzung am Ende einer Zeit T3 zwischen dem Senden jedes Kapitels und einer Zeit T5 zwischen jedem Senden des Rahmens.

Gemäß einem weiteren Merkmal sendet der Server einen Kapitelenderahmen, wenn während einer Zeit T4 nach dem Senden eines Kapitelenderahmens keine andere Station eine Anforderung gesendet hat.

Gemäß einem weiteren Merkmal sind die Zeiten T1 bis T5 parametrisierbar.

Ein weiteres Ziel der Erfindung ist, ein Verfahren zum Fernladen vorzuschlagen, das nach dem Beenden des Fernladens einer ersten Betriebssystemeinheit außerdem verschiedene Anwendungen fernladen kann, entweder auf Anforderung eines Terminals oder aufgrund einer Mehrteilnehmer-Sendung (Multicast).

Dieses Ziel wird dadurch erreicht, daß das Verfahren einen Schritt des Schickens von für die Kapitel spezifischen Datenrahmen enthält, die die Ethernet-Adresse der Empfangsstation enthalten.

Ein weiteres Ziel ist, den Zugriff auf die in das Terminal geladene Datei zu ermöglichen und für die nicht vorhandenen Dateien deren Laden zu ermöglichen.

Dieses Ziel wird dadurch erreicht, daß das Terminal bei der Ausführung eines Dateiöffnungsbefehls die Dateizuordnungstabelle liest; damit den Sektor bestimmt und dann die Sektorbe-

schreibungstabelle liest, um das zu suchende Kapitel zu bestimmen; das Kapitel ausführt, falls das Kapitel im ausgeführten RAM vorhanden ist, und eine Fernladeanforderung zum Server sendet, falls das Kapitel nicht vorhanden ist.

Gemäß einem weiteren Merkmal enthält die Sektorbeschreibungs-tabelle ein Zustandsbyte, das durch ein Bit den Empfang des Kapitels, durch ein weiteres Bit seine Verwendung, durch ein drittes Bit den Status "stets beibehalten", durch ein vierthes Bit die Segmentnummer, durch ein fünftes Bit das Vorhandensein eines Füllwertes und durch ein sechstes Bit einen spezifischen Datensektor angibt.

Gemäß einem weiteren Merkmal enthält das Terminal ein Zu-standsregister der emulierten Unterbrechungsfunktion, die im Fall des Empfangs eines Fernladeenderahmens ermöglicht, durch Lesen der Register (DL und AL) des Mikroprozessors (213) des Terminals zu bestimmen, ob eine Funktion "Datensegment laden" oder eine Schließfunktion ("Shutdown") ausgeführt werden soll.

Weitere Merkmale und Vorteile der vorliegenden Erfindung werden deutlich beim Lesen der folgenden Beschreibung, die mit Bezug auf die beigefügte Zeichnung gegeben wird, worin:

Fig. 1 eine schematische Ansicht der Speichereinheit ist, die die Ausführung des Verfahrens zum Fernladen ermöglicht;

Fig. 2 eine schematische Darstellung des Zeitdiagramms der Übertragungen zwischen dem Server und einer oder mehrerer Stationen ist;

Fig. 3 eine schematische Ansicht ist, die den Zustand des Netzes nach dem Laden des Betriebssystems darstellt;

Fig. 4A bis 4C den Aufbau verschiedener Rahmen darstellen, die in der Fernladeprozedur verwendet werden.

Fig. 1 zeigt einen Server (1), der über eine Telekommunikati-onskarte (10) mit einem Ethernetnetz (3) und über das Ether-

netnetz (3) mit wenigstens einem ersten (ST1) und einem zweiten (ST2) Terminal verbunden ist. Jedes Terminal ST1 (21), ST2 (22) ist in der gleichen Weise durch eine Karte zur Telekommunikation mit dem Ethernetnetz (211) gebildet, auf der sich ein Prozessor (2110) und ein PROM zur Steuerung des Prozessors (2110) befindet. Dieser PROM (2110) enthält neben dem Programm zur Steuerung des Prozessors entweder ein Programm zur Einrichtung der Mehrteilnehmer-Verbindung (Multicast) UDP mit dem Server oder ein Programm zur Einrichtung der Einteilnehmer-  
10 Verbindung (Unicast) IP mit dem Server. Der PROM enthält außerdem ein Programm zur Emulation der Unterbrechung 13 und der Unterbrechung 19. Das Terminal selbst enthält einen Mikroprozessor (213) und einen Schreib-Lese-Speicher RAM mit einer Kapazität von wenigstens 1 MByte, der mit dem Mikroprozessor über den Bus verbunden ist. Der Mikroprozessor (213) kommuniziert über den Bus mit der Telekommunikationskarte (211), und das in dem Nur-Lese-Speicher (ROM) enthaltene Initialisierungsprogramm ermöglicht bei einer Neuinitialisierung das Erzeugen einer Unterbrechung 19, die durch das im Speicher  
20 PROM (2110) gespeicherte Programm emuliert wird. Diese Emulation ermöglicht mit Hilfe der Programme UDP oder IP das Einrichten einer Verbindung mit dem Server. Der Speicher (2110) enthält außerdem ein Protokoll zum Fernladen (PROTCHG) und einen Treiber NDIS. Wenn die Station unter Spannung gesetzt wird, löst das Emulationsprogramm der Unterbrechung (IT19) das Schicken eines für den Server bestimmten Rahmens durch das Kommunikationsprotokoll UDP oder IP aus, und das Fernladeprotokoll des Servers TCHG antwortet auf diesen Rahmen, indem es die Dateizuordnungstabellen (FAT) und die Sektorbeschreibungstabellen (SDT), die in dem Platten-Cache (CD) des Schreib-Lese-Speichers RAM (212) gespeichert sind, zurückschickt. Nach dieser Übertragung und nachdem der Server ein Programm BOOT zur Initialisierung der Station geschickt hat, führt diese das Programm aus, wobei sie, wenn sie auf eine Anforderung zum Lesen einer Datei stößt, die nicht in ihrem Schreib-Lese-Speicher vorhanden ist, eine Unterbrechung INT13 für eine Zugriffsanforderung auf die Magnet-Peripheriegeräte erzeugt, die in der Station nicht vorhanden

sind. Diese Zugriffsanforderung auf die Magnet-Peripheriegeräte löst das Ausführen eines Emulationsprogramms aus, das das Einrichten einer Verbindung mit dem Server (1) auslöst, wobei dieser aufgefordert wird, ein Kapitel zu senden, das dem Namen der Datei, deren Öffnen angefordert worden ist, entspricht. Die Dateizuordnungstabelle (FAT) ermöglicht das Herstellen einer Korrespondenz zwischen dem Dateinamen und der Diskettennummer, der Kopfnummer, der Spurnummer und der Spursektornummer, die den Ort der Datei auf einer Station angeben, die über eine Diskette oder eine Platte verfügt, wobei die Tabelle SDT das Herstellen einer Korrespondenz zwischen der Sektornummer und der Nummer des Kapitels, das von dem Server anzufordern ist, ermöglicht. Im Anschluß an diese Anforderung bereitet der Programmanager (ADM) des Servers in einer Bilddatei (IMA) das durch die Station angeforderte Kapitel vor. In Fig. 1 enthält die Bilddatei Dateien des Betriebssystems MS-DOS, die durch die Dateien IO.SYS, MS-DOS.SYS, CONFIG.SYS und COMMAND.COM gebildet sind. Bei jedem Ausführen des Betriebssystems MS-DOS fordert die Station das Öffnen der für ihren Betrieb erforderlichen Dateien an, wobei dieses Dateiöffnen eine Anforderung eines Kapitels auslöst, bis sämtliche der für den Betrieb des Systems erforderlichen Dateien geladen sind, so daß ein normaler Betrieb des Terminals möglich ist. Unter der Annahme, daß nur das Terminal ST1 in Betrieb ist, entspricht in diesem Moment die Systemkonfiguration der in Fig. 3 gezeigten Konfiguration, in der festgestellt wird, daß der Schreib-Lese-Speicher (212) mit den Programmen IO.SYS, MS-DOS.SYS, CONFIG.SYS, COMMAND.COM, NETFLOP.DOS geladen worden ist, wovon das letzte Programm dazu dient, die Steuerung der Kommunikationskarte in der Weise auszuüben, daß gegebenenfalls auf besondere Anforderungen eines Sektors oder eines Kapitels ein späteres Fernladen möglich ist. Das ausgeführte Protokoll zum Fernladen ist in Fig. 2 genauer dargestellt, in der zu sehen ist, daß die Station (1) eine Initialisierungsanforderung (BD) sendet, auf die der Server mit einer Nachricht "BOOT REPLY" (BR) antwortet. Im Anschluß an diese Antwort sendet die Station eine Anforderung eines Kapitels, indem sie im Fall des

ersten Kapitels einen Rahmen "starte Kapitel 1" (SC1) sendet, auf die der Server im Fall einer Mehrteilnehmer-Kommunikation mit der Anforderung (ARPD) einer Adresse in dem Ethernetnetz antwortet, worauf die Station diese Informationen durch die Antwort (ARPR) liefert. Im Anschluß an diese Antwort sendet der Server einen Rahmen (SDTC1) mit der Sektorbeschreibungstabelle des Kapitels 1; dann sendet er nach Ablauf einer Zeit T5 die Daten des Kapitels 1 (DC1), die in mehreren Rahmen, zwischen denen stets ein Zeitintervall T5 vorhanden ist, geschickt werden können. Der Server wartet eine Zeit T1 zwischen der Nachricht "start chapter 1" (SC1), die den Beginn des Fernladens anfordert, und der Antwortnachricht, die die Adresse anfordert. Dieses Zeitintervall ermöglicht anderen Stationen, deren Hochfahren zwischenzeitlich geschehen kann, ebenfalls ihre "start chapter 1"-Nachrichten (SC1) zu senden, die sofort durch den Server übernommen werden. Am Ende des Sendens des Kapitels 1 sendet die Station, falls sie einen Fehler feststellt, eine Nachricht SDTE über einen Fehler in einem Teil des Kapitels 1, worauf der Server in diesem Fall den Rahmen SDT des Kapitels 1 (SDTC1) und die spezifischen Daten dieses Kapitels (SDC1), die nicht empfangen worden sind, zurückschickt. Zum Schluß schickt der Server einen Schlußrahmen des Kapitels 1 "end chapter 1" (EC1), wobei die Station dieses empfangene Kapitel 1 durchführt, um im Verlauf des Ausführens festzustellen, daß die dem empfangenen Kapitel entsprechende Datei den Aufruf einer weiteren Datei verlangt. Dies stößt eine Anforderung eines weiteren Kapitels n durch einen Rahmen SCn "start chapter n" an. Wenn der Server diese Anforderung empfängt, wartet er eine Zeit T3, bevor er den Rahmen SDTCn, der aus der Sektorbeschreibungstabelle des Kapitels n besteht, und den Rahmen DCn mit den Daten dieses Kapitels zurückschickt. Wenn wie oben die Station einen Rahmen über einen Fehler in dem Kapitel n (SDTECn) sendet, schickt der Server den Rahmen mit der Sektorbeschreibungstabelle des Kapitels (SDTCn) und die spezifischen Daten für diese Kapitel SDCn, die schlecht empfangen worden sind, zurück. Wenn die Station den Schlußrahmen des Kapitels n (ECn) empfängt, wird das empfangene Kapitel ausgeführt, während im Fall eines

Fehlers in den Daten ein Rahmen über einen Fehler in den Daten des Kapitels n (DECn) gesendet wird, um von Seiten des Servers das Senden eines Rahmens mit den Daten des Kapitels n (DCn) zu erlangen. Das Senden wird erneut durch einen Schlußrahmen des Kapitels n (Cn) beendet. Der Server wartet anschließend eine Zeit T4, während der er, wenn er keine gegenteilige Anforderung erhält, einen Rahmen "end remote boot" (ERB), d. h. Ende des Fernladens, sendet. Falls der Server mit mehreren Stationen verbunden ist und eine zweite Station 2 ST2 eine Anforderung eines ersten Kapitels SC1 sendet, indem sie sich während eines Zeitintervalls T2 im Anschluß an die Verbindung mit einer ersten Station an den Speicher schaltet, wird die Anforderung dieser zweiten Station sofort übernommen, wobei die Prozedur des Fernladens der ersten Station annulliert wird und die Prozedur für die Gesamtheit der beiden Stationen wieder bei null begonnen wird. Dieses Verfahren hilft, zu vermeiden, daß, um eine zweite Station 2 fernzuladen, auf das Ende des Fernladens einer Station 1 gewartet werden muß, oder zu vermeiden, daß auf der Serverebene ein Speicher, der die nicht empfangenen Kapitel verwaltet, am Ende des Fernladens die restlichen zu empfangenen Kapitel anfordern muß, wie es beim US-Patent 4 958 278 der Fall ist.

Jeder im Verlauf der in Fig. 2 dargestellten Fernladesequenz gesendete Rahmen hat einen spezifischen Aufbau, der dem in den Fig. 4A bis 4C dargestellten Aufbau entspricht. So ermöglicht der Rahmen "starte Kapitel" (SC) durch die zwei ersten Bytes, den Rahmentyp anzugeben und dem Server mitzuteilen, daß es sich um einen Rahmen "starte Kapitel" handelt, der das Senden eines Kapitels anfordert, durch die zwölf folgenden Bytes den Namen des Bandes, in dem sich das zu schickende Kapitel befindet, zu definieren, durch die zwei folgenden Bytes den Identifizierer, d. h. die Kapitelnummer im ASCII-Code, zu definieren und durch die folgenden zwei Worte die "Ethernet"-Adresse (IP-Station) der Station zu definieren.

Im weiteren werden die Informationen, die in den anderen Rahmen identisch und vom gleichen Format sind, nicht nochmals beschrieben.

Der Rahmen SDTC "Sektorbeschreibungstabelle des Kapitels" enthält weitere Informationen: den Rahmentyp, den Bandnamen, den Identifizierer, die Informationen, die aus einem Byte gebildete Rahmenmaske definieren und durch die Bits A1 den Namen des Rahmens, der zu dieser Formation SDTC gehört, angeben. Ein zusätzliches Byte definiert die Maximummaske des Rahmens, in dem alle Bits auf "1" gesetzt sind, die folgenden zwei Bytes definieren das erste Datensegment und die zwei weiteren, folgenden Bytes definieren das zweite Datensegment, falls dieses vorhanden ist. Zwei zusätzliche Bytes definieren den relativen Abstand, in dem die folgenden Karten der Sektorbeschreibungstabelle eingefügt sind, und zwei zusätzliche Bytes definieren die Anzahl der folgenden Karten. Die Karten sind jeweils durch 8 Bytes definiert und enthalten, wie durch die Karte SDC dargestellt ist, in zwei Bytes die Sektornummer, in zwei weiteren Bytes das Pseudonym des Sektors, in den zwei folgenden Bytes den relativen Abstand in dem Disketten-Cache, in einem Byte den Füllwert und in einem weiteren Byte den Zustand der Karte, der definiert ist durch: Bit 7, das, wenn es gleich dem logischen Wert "1" ist, angibt, daß der Sektor empfangen worden ist, Bit 6, das, wenn es gleich "1" ist, angibt, daß der Sektor verwendet worden ist, Bit 5, das, wenn es gleich "1" ist, angibt, daß der Sektor auf Dauer zu speichern ist, Bit 4, das, wenn es gleich "1" ist, die Segmentnummer anzeigt, Bit 3, das, wenn es gleich "1" ist, einen Füllwertes anzeigt, und Bit 0, das, wenn es gleich "1" ist, für den Sektor spezifische Daten anzeigt.

Der Rahmen "SDT-Fehler" (SDTE) ist wie die folgenden Rahmen aus Informationen aufgebaut, die den Rahmentyp, den Bandnamen, den Identifizierer, die "Ethernet"-Adresse der Station und die Maske des Rahmens SDTC betreffen. Der Rahmen "Kapitel mit spezifischen Daten" (SDC) enthält ebenfalls die Informationen: Rahmentyp, Bandnamen, Identifizierer, relativer Abstand der ersten Karte, relativer Abstand der zehnten Karte, in 6 Bytes eine Ethernetadresse, die die Adresse der Station definieren, und in 512 bis 1024 Bytes die zu sendenden Daten.

Der Rahmen "Datenfehler" (DE) enthält im weiteren den Rahmentyp, den Bandnamen, den Identifizierer, in 2 Bytes Informationen, die die Anzahl der fehlenden Sektoren angeben, und in 2 weiteren Bytes für jeden fehlenden Sektor die Nummer des fehlenden Sektors.

Der Rahmen EC "Ende des Kapitels" und der Rahmen für das Ende der Initialisierung "End remote boot" (ERB) enthalten den Rahmentyp und den Bandnamen.

Der Rahmen SSD "Anforderung des spezifischen Sektors" enthält neben dem Rahmentyp den Bandnamen, den Identifizierer und in 2 Bytes die Nummer des angeforderten spezifischen Sektors.

Der Rahmen SSR enthält im weiteren Informationen, die den Daten in 512 Bytes vorhergehen.

Das Programm für die Emulation der Unterbrechung 13 modifiziert durch Umweg über das Zustandsregister der emulierten Funktionen die in den Registern des Stationsprozessors (213) enthaltenen Informationen gemäß der im folgenden angegebenen Tabelle:

0	reset	Zurücksetzen des Merkers "carry" (NY)
1	disk_status	Lesen des Diskettenzustands (Funktion "read") und Zurücksetzen des Merkers "carry" (NY)
2	read	siehe Beschreibung weiter unten
3	write	Rücksprungcode "write protect" und Merker "carry" bei Fehler (CY)
4	verify	Zurücksetzen des Merkers "carry" (NY)
5	format	Setzen des Merkers "carry" bei Fehler (CY)
8	parameters	Lesen der Diskettenparameter und Zurücksetzen des Merkers "carry" (NY)
d	special	spezielles Fernladen. (siehe "Status von TCHG nach Empfang von "end_remote_boot - mit Netflop")

- wenn die Register DL = 0 und AL = 0 sind, handelt es sich um die Funktion "change Data Segment" zwischen dem PROM und Netflop
- wenn die Register DL = 0 und AL = 1 sind, handelt es sich um die Funktion "shutdown" des PROM

15 read_dasd	Rücksprungcode "change_line avail." und Zurücksetzen des Merker "carry" (NY)
16 change_line	Rücksprungcode "non actif" und Zurücksetzen des Merkers "carry" (NY)
17 set_dasd	Rücksprungcode OK und Zurücksetzen des Merkers "carry" (NY)
18 set_media	Rücksprungcode " write protect" und Merker "carry" bei Fehler (CY)

Alle anderen Funktionscodes geben eine Rücksprungcode 01 zurück.

Somit kann festgestellt werden, daß in dem Fall der speziellen Funktion, die durch den von 9 bis 15 enthaltenen Code dargestellt ist, die Register DL und AL des Mikroprozessors mit den logischen Werten geladen werden, die entweder einen Austausch von Datensegmenten oder die Schließfunktion des PROM "shutdown" angeben. Die anderen emulierten Funktionen ermöglichen die Modifikation der Bits (CY) und (NY) des Zustandsregisters des Mikroprozessors.

Wenn die in Fig. 3 dargestellte Datei CONFIG.SYS, die die Dateien PROTMAN.DOS, ELNKII.DOS, NEMM.DOS und TCPDRV.DOS umfaßt, die die Funktionen des Protokollmanagements, des Kartentreibers, des Speichermanagements bzw. der Managements des residenten Teils des Programms garantieren, und das gesamte Betriebssystem sowie das Programm NETBIND.EXE geladen worden sind, sendet der Server einen Rahmen für das Ende der Verwendung "end remote boot" (ERB), durch den er dem PROM anzeigt, daß die Gesamtheit der Programme geladen worden ist, worauf das im PROM enthaltene Protokoll zum Fernladen dieses Fernladens durch mehrere Aktionen beendet, die darin bestehen, das Ausführen des Programms NETBIND anzustoßen, welches das

Programm ELNKII.DOS verwendet, um auf den Speicher zuzugreifen, wobei dieses Programm NETBIND das Ausführen des Programms NETFLOP anstößt, welches das Laden des Datensegments ermöglicht, und den Kontext der durch den PROM gesteuerten Sitzung wiederherstellt. NETFLOP führt außerdem ein normales Schließen des PROM (Shutdown) mit einem Rücksetzen der Speicherkarte und der Restaurierung des Vektors IRQ5 auf den ursprünglichen BIOS-Vektor sowie ein Außerbetriebsetzen des Steuerprogramms für die programmierbaren Verweilzeiten des Fernladeprozesses aus. Dieses Schließen ruft den Anstoß für eine Steuerung der Unterbrechungen und vor allem der Unterbrechung 13 des PROM in bezug auf das Programm NETFLOP hervor. Mit dieser neuen Steuerung kann über einen parametrierbaren Ablauf verfügt werden, der von dem starren Ablauf des PROM verschieden ist. Die Steuerung des INT13 durch den PROM ist gültig, solange das Schließen (Shutdown) nicht ausgeführt ist. Nach dem Schließen ruft die Unterbrechung 13 des PROM entweder den Unterbrechungsvektor 13 von NETFLOP oder den ursprünglichen Vektor BIOS auf. Dieses Schließen wird durch das Ausführen der im folgenden beschriebenen Sequenz durch einen Mikroprozessor (213) des Typs INTEL 386 oder 482 ausgeführt, wobei die Umtragung auf andere Mikroprozessoren im Bereich des Wissens des Fachmanns liegt.

```
Floppy proc near
    push ds - Schieben des Datensegments ds in den
              Stapel des Registers
    push ax - Schieben der Komponente ax in den Stapel
              der Register AH und AL
    xor ax,ax - Exklusiv-ODER über ax
    mov ds,ax Verschieben von ax -> ds
    mov ax,[word ptr ds:[4a2h]] - Verschieben des Inhalts
                                der Adresse, die durch den Inhalt von ds + 4a2h
                                in ax bestimmt ist
    mov ds,ax Verschieben von ax in ds
    cmp di,0 Vergleichen des Registers di mit dem
                  Wert 0
    pop ax Auszug des Registers ax
```

jne not\_ours, wenn di ungleich ZF = 0, Sprung auf die Marke not ours  
cmp ah,19h Vergleich von 19h mit dem oberen Abschnitt von ax  
jae not\_ours, Sprung auf "not ours", wenn ah kleiner als 19h => CP = 0  
test boot\_state, SHUTDOWN  
jnz another\_story, Sprung auf "another\_story", wenn ungleich 0 => ZF = 0 (wenn boot\_state = SHUTDOWN)  
push es - Schieben in den Stapel des Registers es  
pusha Sichern der 8 allgemeinen Register in dem Stapel  
mov bp,sp - Verschieben von SP --> BP  
mov si,ax - Verschieben von AX --> SI  
shr si,8 - logisches Schieben um 8 nach rechts; get rid of AL  
shl si,1 - Verschieben um 1 nach links;  
call cs: Dispatch[si] - Abruf des Inhalts von Dispatch, das mit si indiziert ist  
call store\_status - Aufruf der Routine store\_status  
popa Extraktion der 8 allgemeinen Register des Stapels  
pop es - Extraktion des Registers es  
pop ds - Extraktion des Registers des Datensegments  
iret return form interrupt

not\_ours

pushf Schieben der Merker in den Stapel  
call dword ptr save\_13; ursprünglicher Vektor (BIOS)  
pop ds - Extraktion des Datensegments  
retf 2 ; wegen Merker "carry"

another\_story

pushf  
call dword ptr new\_it\_13; Vektor von Netflop  
pop ds  
retf 2 ; wegen Merker "carry"

floppy endp

Die Steuerung des INT13 durch NETFLOP ruft die in einer Datei PROTOCOL.INI deklarierten Parameter "KEEP FLOPPY" und "KEEP CACHE" auf, die abhängig von ihren Zuständen die folgenden Funktionen auslösen:

KEEPCACHE = YES

KEEPFLOPPY = YES

- INT13h      Suche den Sektor in dem Disketten-Cache. Wenn der Sektor sich nicht oder nicht mehr darin befindet, wird eine Anforderung "specific sector" in Richtung des Servers TCHG erzeugt. Dieser greift direkt auf die Diskette (diejenige, deren FAT und Hauptverzeichnis zur Station geschickt worden sind, muß zwingend in das Lesegerät des Servers eingegeben werden) und antwortet der Station durch einen Rahmen "specific sector reply" (SSR), der im Mehrteilnehmer-Ethernet adressiert ist, jedoch in dem Protokollabschnitt des TCHG die Adresse IP der Zielstation enthält. Die Stationen filtern dieses Feld nach Empfang eines solchen Rahmens.

Somit bleibt nach Beendigung des Fernladens ein Mini-Dateimanagement für die Station verfügbar.

KEEPCACHE = YES

KEEPFLOPPY = NO

- INT13h      Suche den Sektor in dem Disketten-Cache. Wenn der Sektor sich nicht oder nicht mehr darin befindet, wird dem System ein Rücksprungcode "beim Lesen kein Zugriff auf den Sektor" mitgeteilt.

- Der Cache bleibt im Speicher aktiv.

KEEPCACHE = NO

KEEPFLOPPY = YES

- INT13h      Eine Anforderung "specific sector" wird in Richtung des Servers TCHG erzeugt. Dieser greift direkt auf die

Diskette (diejenige, deren FAT und Hauptverzeichnis zur Station geschickt worden sind, muß zwingend in das Lesegerät des Servers eingegeben werden) und antwortet der Station durch einen Rahmen "specific sector reply", der im Mehrteilnehmer-Ethernet adressiert ist, jedoch in dem Protokollschnitt des TCHG die Adresse IP der Zielstation enthält. Die Stationen filtern dieses Feld nach Empfang eines solchen Rahmens.

Somit bleibt nach Beendigung des Fernladens ein Mini-Dateimanagement für die Station verfügbar (siehe die mit dem Wechseln von Disketten verbundenen Leistungsmerkmale und Einschränkungen).

- Der Cache ist inaktiv, wobei der Speicher somit für die Anwendungen freigegeben ist.

KEEP CACHE = NO

KEEP FLOPPY = NO

- INT13h Dem System wird ein Rücksprungcode "beim Lesen kein Zugriff auf den Sektor" mitgeteilt.

- Der Cache ist inaktiv, wobei der Speicher somit für die Anwendungen freigegeben ist.

Der Parameter "KEEP CACHE=YES" bedingt, daß der durch den Cache verwendete Speicherbereich nicht Gefahr läuft, mit einer DOS-Anwendung in Konflikt zu kommen.

#### Beispiel:

Wenn das Segment CACHE mit 7000H gewählt ist, überschreibt eine Anwendung, die dieses Segment verwendet, die Daten des Cache, was eine große Gefahr des Blockierens des Systems mit sich bringt.

Umgekehrt bleiben die Segmente 0D000H zu jeder Zeit für den Cache verfügbar (mit Ausnahme jeder Verwendung für das Spei-

cherseitenmanagement (EMS), das sehr wohl denselben Speicherbereich benutzt).

Anmerkung: Die Segmente 0D000H und/oder 0E000H werden, wenn sie verwendet werden, nach dem Empfang von "end\_remote\_boot" nicht freigegeben, d. h. genauer:

- mit dem Empfang der SDT (Sektorbeschreibungstabelle) des Kapitels 1 nach dem Erkennen der Arbeitssegmente 0D000H und/oder 0E000H werden diese Segmente validiert.
- mit dem Empfang von "end\_remote\_boot" werden diese Segmente nicht validiert.

0 538 133 (92402820.2)

### Patentansprüche

1. Verfahren zum Fernladen wenigstens eines Terminals durch einen Server, wobei das Terminal einen flüchtigen Speicher (RAM), wenigstens einen Prozessor und eine Telekommunikationskarte enthält, die den Anschluß an den Server des Netzes ermöglicht, wobei der Server mit Telekommunikationsmitteln und mit Speichermitteln mit einer Kapazität versehen ist, die ausreicht, um das Kommunikationsprotokoll, das Betriebssystem der Terminals, das in eine Bilddatei kopiert ist, sein eigenes Betriebssystem und die Anwendungsprogramme zu speichern, dadurch gekennzeichnet, daß es eine Prozedur zum Fernladen eines Ureingabeprogramms enthält, die darin besteht, die erste Unterbrechung zum Lesen des Ureingabeblocks auf der lokalen Platte, die von einem Terminal erzeugt wird, wenn es unter Spannung gesetzt wird, in eine Anforderung zum Lesen einer im Server gespeicherten Bilddatei umzusetzen, wobei das Programm zum Umsetzen von Unterbrechungen (EMUL INT) in einem Festwertspeicher (PROM) der Kommunikationskarte gespeichert ist.
2. Verfahren nach Anspruch 1, dadurch gekennzeichnet, daß der Schritt des Fernladens des Ureingabeprogramms darin besteht, in den flüchtigen Speicher RAM den Ureingabesektor (BOOT) zu laden, die Ausführung dieser Ureingabedatei zu starten und die Unterbrechungen der Zugriffsanforderungen auf Dateien von Magnet-Peripheriegeräten, die nicht im Terminal vorhanden sind, in Anforderungen zum Lesen der entsprechenden Bilddatei im Server durch eine Koordinatentransformationsprozedur umzusetzen.
3. Verfahren nach Anspruch 2, dadurch gekennzeichnet, daß die Koordinatentransformationsprozedur darin besteht, die Koordinaten physischer Sektoren von Dateien von Magnet-Peripheriege-

räten in eine logische Kapitelnumerierung der Bilddatei, die dem Sektor der aufgerufenen Datei entspricht, durch eine Sektorentsprechungstabelle umzusetzen, die Anforderung zum Lesen des Kapitels an den Server zu übertragen und die Kapitel der Bilddateien an sämtliche Terminals zu senden.

4. Verfahren nach einem der Ansprüche 1 bis 3, dadurch gekennzeichnet, daß der flüchtige Speicher RAM des Terminals nach dem ersten Fernladen des Ureingabeprogramms eine Dateizuordnungstabelle (FAT), die eine Korrespondenz zwischen der Diskettennummer, der Kopfnummer, der Spurnummer, der Spursektronummer und dem Dateinamen herstellt, sowie eine Sektorbeschreibungstabelle (SDT), die eine Entsprechung zwischen der Diskettennummer, der Spurnummer, der Spursektronummer und der Kapitelnummer der Bilddatei des Servers herstellt, enthält.

5. Verfahren nach einem der vorangehenden Ansprüche, dadurch gekennzeichnet, daß es das Fernladen von Dateien des Betriebssystems ermöglicht, das durch die Dateien IO.SYS, MSDOS.SYS, CONFIG.SYS und COMMAND.COM gebildet ist.

6. Verfahren nach Anspruch 4, dadurch gekennzeichnet, daß die Dateizuordnungstabelle (FAT) mit dem Dauerstatus (KEEP FOR EVER) im flüchtigen Speicher RAM gespeichert ist, wobei dieser Status durch ein Bit des Zustandsworts der Sektorbeschreibungskarte angegeben wird.

7. Verfahren nach einem der vorangehenden Ansprüche, dadurch gekennzeichnet, daß der Server dann, wenn ein Terminal eine Anforderung eines ersten Kapitels zu diesem Server schickt, eine Zeit T1 vor seiner Antwort abwartet, um sicherzustellen, daß die anderen Terminals nicht die gleiche Anforderung haben.

8. Verfahren nach einem der vorangehenden Ansprüche, dadurch gekennzeichnet, daß eine Anforderung des ersten Kapitels durch eine andere Station, die vor dem Ablauf einer Zeit T2 nach der Anforderung desselben ersten Kapitels durch eine erste Station ankommt, sofort berücksichtigt wird, wobei die Fernladeproze-

dur für die Gesamtheit der Stationen, die Anforderungen gesendet haben, für diese Stationen auf null zurückgesetzt wird.

9. Verfahren nach einem der vorangehenden Ansprüche, dadurch gekennzeichnet, daß der Server eine Zeit T3 zwischen dem Senden jedes Kapitels und eine Zeit T5 zwischen jedem Senden des Rahmens wartet.

10. Verfahren nach einem der vorangehenden Ansprüche, dadurch gekennzeichnet, daß der Server die Sitzung am Ende einer Zeit T4 nach dem Senden eines Kapitelenderahmens schließt, wenn keine andere Station eine Anforderung gesendet hat.

11. Verfahren nach einem der Ansprüche 7 bis 9, dadurch gekennzeichnet, daß die Zeiten T1 bis T5 parametrisierbar sind.

12. Verfahren nach einem der Ansprüche 1 bis 6, dadurch gekennzeichnet, daß es einen Schritt des Schickens von für die Kapitel spezifischen Datenrahmen, die die Ethernetadresse der Empfangsstation enthalten, enthält.

13. Verfahren nach Anspruch 4, dadurch gekennzeichnet, daß das Terminal bei der Ausführung eines Dateiöffnungsbefehls die Dateizuordnungstabelle liest; damit den Sektor bestimmt und dann die Sektorbeschreibungstabelle liest, um das zu suchende Kapitel zu bestimmen; das Kapitel ausführt, falls das Kapitel im ausgeführten RAM vorhanden ist, und eine Fernladeanforderung zum Server sendet, falls das Kapitel nicht vorhanden ist.

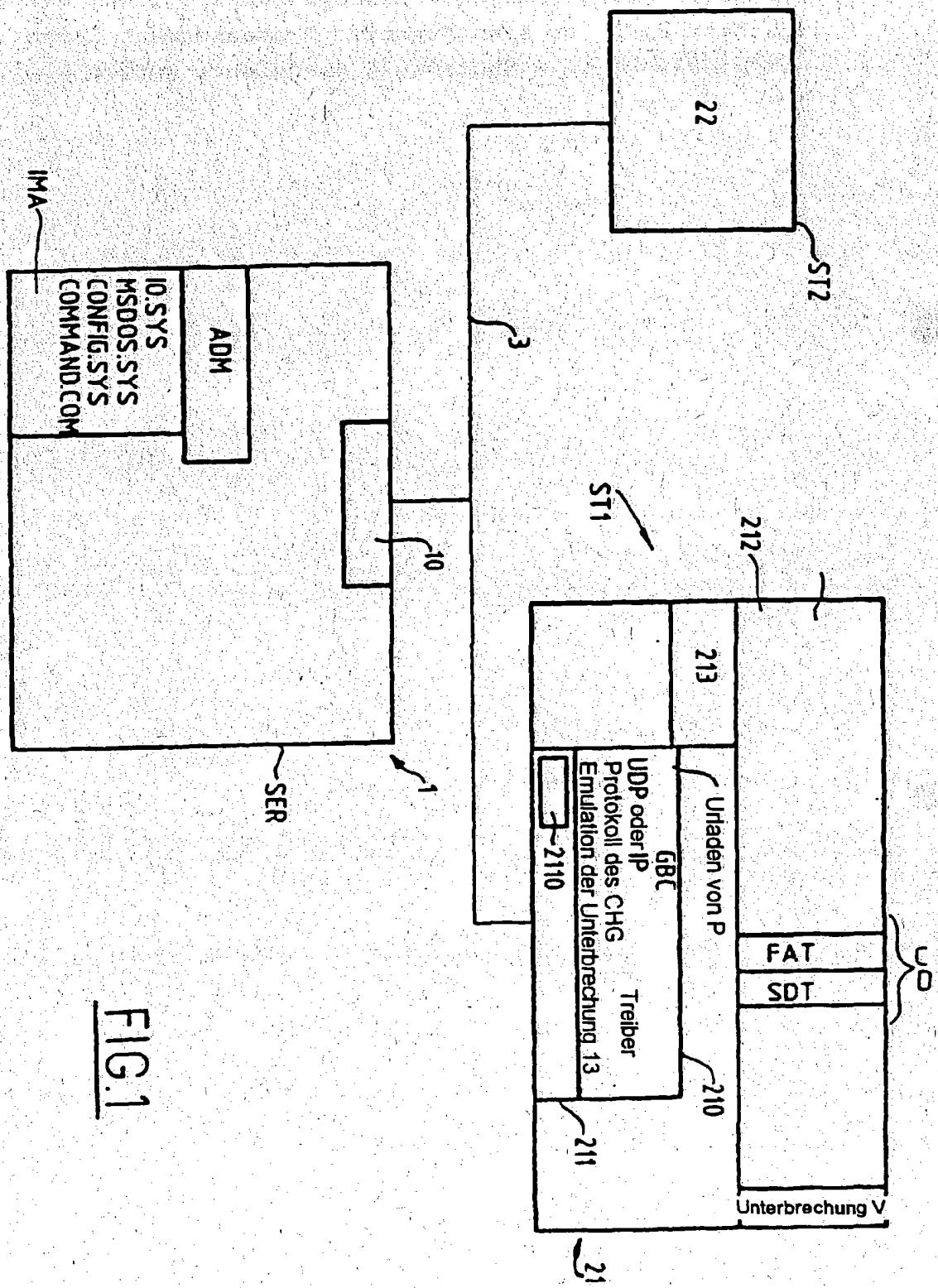
14. Verfahren nach Anspruch 4, dadurch gekennzeichnet, daß die Sektorbeschreibungstabelle ein Zustandsbyte enthält, das durch ein Bit den Empfang des Kapitels, durch ein weiteres Bit seine Verwendung, durch ein drittes Bit den Status "stets beibehalten", durch ein viertes Bit die Segmentnummer, durch ein fünftes Bit das Vorhandensein eines Füllwertes und durch ein sechstes Bit einen spezifischen Datensektor angibt.

15. Verfahren nach einem der vorangehenden Ansprüche, dadurch gekennzeichnet, daß das Terminal ein Zustandsregister der

19.01.99

- 22 -

emulierten Unterbrechungsfunktion enthält, die im Fall des Empfangs eines Fernladeenderahmens ermöglicht, durch Lesen der Register (DL und AL) des Mikroprozessors (213) des Terminals zu bestimmen, ob eine Funktion "Datensegment laden" oder eine Schließfunktion ("Shutdown") ausgeführt werden soll.

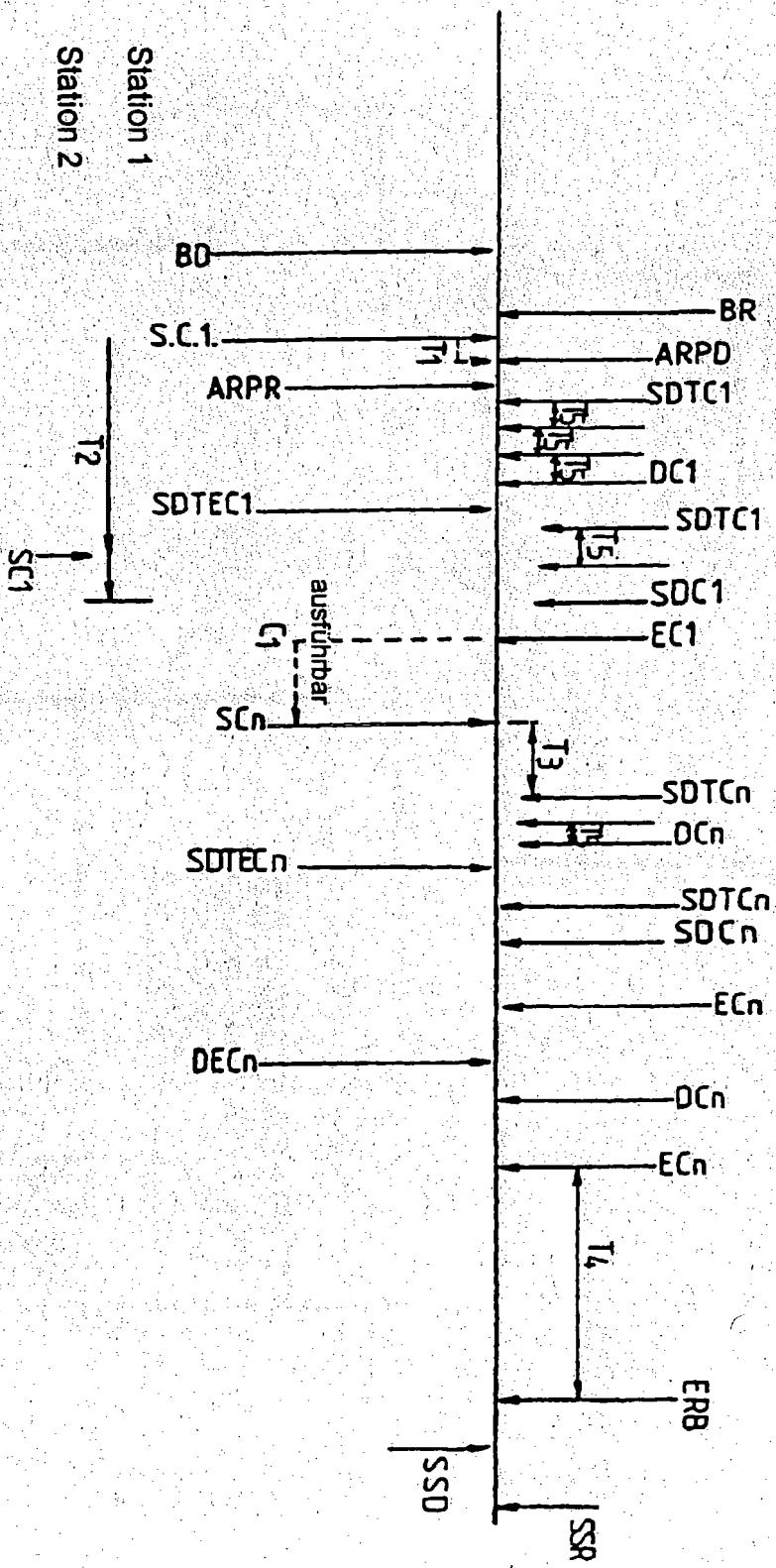


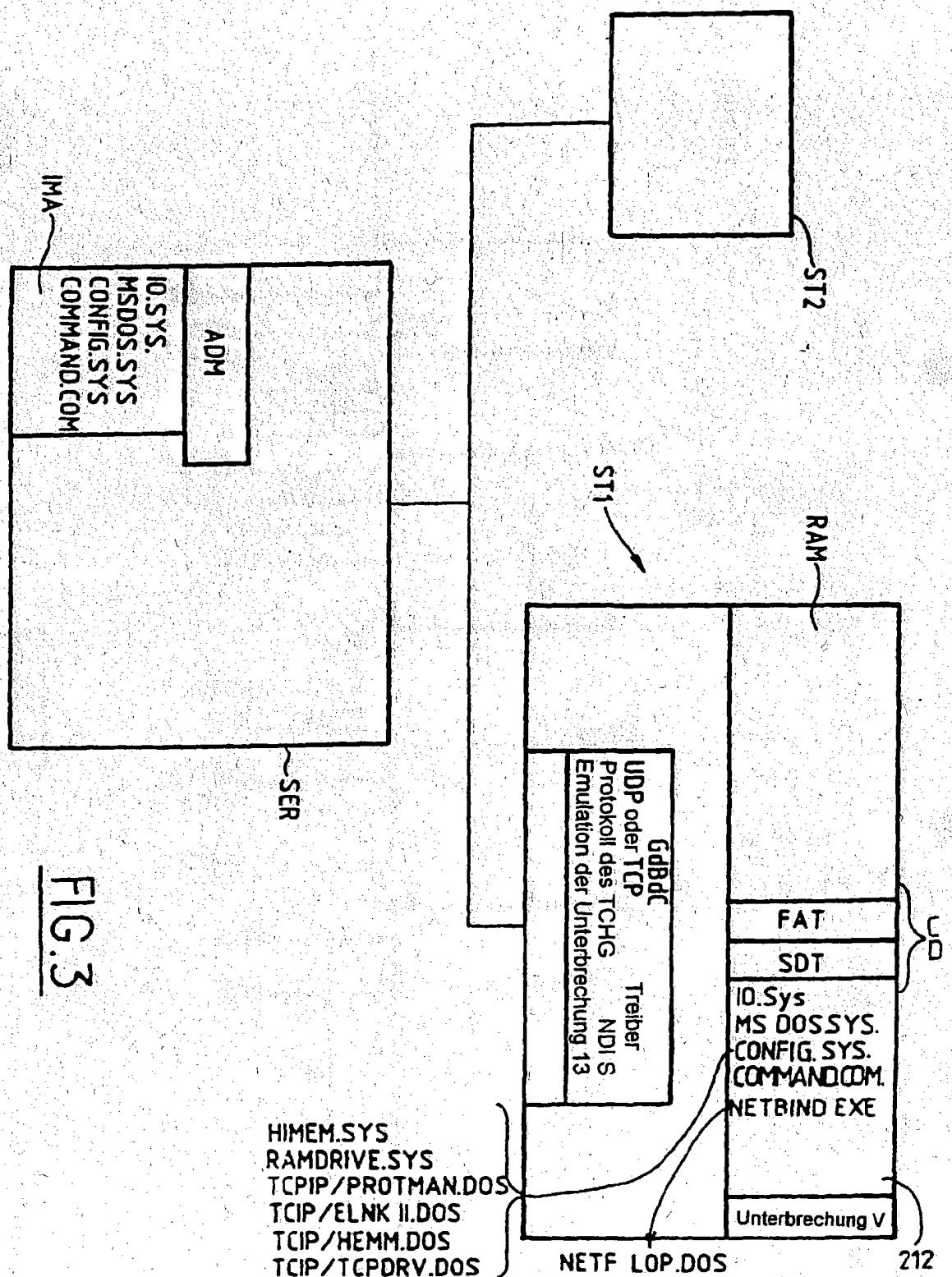
19.01.98

- 24 -

Server

FIG. 2





**SC -Startkapitel**

Rahmentyp	2 Bytes	000H
Bandname	12 Bytes	
Identifizierer	2 Bytes	gibt die Nr. des Kapitels in ASCII an (01 für das 1.)
IP-Station	2 Wörter	

**SDTC - Sektorbeschreibungstabelle des Kapitels**

Rahmentyp	10 Bytes	100H
Bandname	12 Bytes	
Identifizierer	2 Bytes	Nr. des Kapitels
Maske des Rahmens SDTC	1 Byte	1 Bit mit "1" pro Rahmen
Maske des maximalen Rahmens SDTC	1 Byte	1 Bit mit "1" pro Rahmen
1. Datensegment	2 Bytes	0, wenn keines vorhanden
2. Datensegment	2 Bytes	
Versatz, mit dem die folgenden Karten von SDT angeordnet sein müssen.	2 Bytes	
Anzahl der folgenden Karten	2 Bytes	
Karten	8 Bytes	pro SDC

**SDC-Karte (Sektorbeschreibungskarte)**

Sektornummer	2 Bytes	
Pseudonym des Sektors	2 Bytes	(0, wenn keine vorhanden)
Versatz	2 Bytes	Versatz im Cache
Diskette (durch die Anzahl der Bits des Segments angegebenes Segment)		
Füllwert	1 Byte	
Zustand	1 Byte	

7	6	5	4	3	2	1	0	
X-----								empfängen
--X-----								nicht verwendet
---X-----								ständig zu speichern
----X----								Segmentnummer
-----X---								Füllwert
-----X-----X								spezifischer Datensektor

**Fig. 4A**

<b>SDTE - SDT-Fehler</b>			
Rahmentyp	2 Bytes		200H
Bandname	12 Bytes		
Identifizierer	2 Bytes		
IP-Station	2 Wörter		
Maske des SDTC-Rahmens	1 Byte		
<b>SDC - Kapitel mit spezifischen Daten</b>			
Rahmentyp	2 Bytes		401H
Bandname	12 Bytes		
Identifizierer	2 Bytes	Nr. des Kapitels nach ASCII	
Versatz der 1. Karte	2 Bytes		
Versatz der 2. Karte	2 Bytes	(0, wenn keine vorhan- den)	
Ethernet-Adresse	6 Bytes		
Daten	512 oder 1024 Bytes		
<b>DE - Datenfehler</b>			
Rahmentyp	2 Bytes		800H
Bandname	12 Bytes		
Identifizierer	2 Bytes	Nr. des Kapitels in ASCII	
Anzahl der fehlenden Sektoren	2 Bytes		
Nummer der Sektoren	2 Bytes	für jeden Sektor	
<b>EC - Kapitelende</b>			
Rahmentyp	2 Bytes		1000H
Bandname	12 Bytes		
Identifizierer	2 Bytes		

Fig. 4B

190109

- 28 -

**ERB - Ende der Ferninitialisierung (end remote boot)**

Rahmentyp	2 Bytes	2000H
Bandname	12 Bytes	
Identifizierer	2 Bytes	Nr. des Kapitels in ASCII

**SSD - Anforderung eines spezifischen Sektors**

Rahmentyp	2 Bytes	4000H
Bandname	12 Bytes	
Identifizierer	2 Bytes	Nr. des Kapitels
Sektornummer	2 Bytes	

**SSR - Antwort eines spezifischen Sektors**

Rahmentype	2 Bytes	4001H
Bandname	12 Bytes	
Identifizierer	2 Bytes	Nr. des Kapitels
Sektornummer	2 Bytes	
Daten	512 Bytes	

**Fig. 4C**

